Bayesian sparse topical coding

Bayesian Sparse Topical Coding

Min Peng, Qianqian Xie, Hua Wang, Yanchun Zhang, Gang Tian

Abstract—Sparse topic models (*STMs*) are widely used for learning a semantically rich latent sparse representation of short texts in large scale, mainly by imposing sparse priors or appropriate regularizers on topic models. However, it is difficult for these *STMs* to model the sparse structure and pattern of the corpora accurately, since their sparse priors always fail to achieve real sparseness, and their regularizers bypass the prior information of the relevance between sparse coefficients. In this paper, we propose a novel Bayesian hierarchical topic models called *Bayesian Sparse Topical Coding with Poisson Distribution* (*BSTC-P*) on the basis of *Sparse Topical Coding with Sparse Groups* (*STCSG*). Different from traditional *STMs*, it focuses on imposing hierarchical sparse prior to leverage the prior information of relevance between sparse coefficients. Furthermore, we propose a sparsity-enhanced *BSTC*, *Bayesian Sparse Topical Coding with Normal Distribution* (*BSTC-N*), via mathematic approximation. We adopt superior hierarchical sparse inducing prior, with the purpose of achieving the sparsest optimal solution. Experimental results on datasets of Newsgroups and Twitter show that both *BSTC-P* and *BSTC-N* have better performance on finding clear latent semantic representations. Therefore, they yield better performance than existing works on document classification tasks.

Index Terms—Document Representation, Bayesian Topic Model, Sparse Coding, Hierarchical Prior.

1 Introduction

SHORT texts have been an important information source for many internet users with the rapid development of social media, such as Facebook, Weibo, and Twitter. These short texts are characterized by fast spreading, short length, large amount, sparse information, snarled noise and irregular modality [16]. Because of these features, short texts cannot be handled by manual and traditional tools. Thus, there is an urgent need for powerful tools which are capable of extracting useful and meaningful latent representations from a large scale of short texts. The extracted latent representations are important to follow-up research and engineering applications, such as emergencies detection [20], [13], user interest modeling [21], Micro-Blogger influence analysis [22], and automatic query-reply [58]. Latent Dirichlet Allocation (*LDA*) [1], [15] is such a useful probabilistic tool for analyzing overwhelming texts. It has made dramatically advances in many domains, such as hyperlink analysis [47]and query processing [50]. LDA-based probabilistic topic models [2], [3], [4] learn the latent document and topic representations from original corpus without any labels. In essence, they generate the latent representations by implicitly capturing the document-level word co-occurrence information [4]. They have been widely used for long texts which have abundant word co-occurrence information for learning in recent years. Nevertheless, short texts are characteristic of short document length, a very large vocabulary, a broad range of topics, and snarled noise. Therefore, the word co-occurrence information in each short document becomes much sparser, inevitably compromising the performance of these models. It also has been proven that when the average token number N of texts is too small, LDA-based probabilistic topic models can not learn topics accurately and may produce mostly incoherent topics [57]. This is because the topics learned from these models are formally a multinomial distribution over N co-occurrent words scattered in different short texts. Table 1 shows examples of topic words by directly applying *LDA* on the Tweets collection. Obviously, the topic words learned by *LDA* are meaningless and incoherent. The words *lauder*, *facebook* and *president* in topic 2 actually belong to different topics, but they are unreasonably allocated to the same topic. To sum up, directly applying these models on short texts will suffer from the severe data sparsity problem.

This problem has gained so much attention that many methods have been made recently, which can be summarized as following two aspects: using sparse priors [9], [8], [16] and imposing posterior regularizations [11], [12], [23]. Both methods are based on the observation that the latent representations to be learned are highly sparse (i.e. each document focuses on a few topics, and each topic focuses on a few words) [12]. The first method introduces auxiliary variables into probabilistic topic models, like "Spike and Slab", but it is ineffective in controlling the sparsity for the admixing proportions of probabilistic topic models (*PTMs*). The second method imposes posterior regularization on non-probabilistic models, like lasso [10], group lasso [24], [6], and sparse group lasso [5]. They can realize sparse representations in real sense. These two attempts are capable of finding full sparse representations [14], [16], which are more interpretable, clear and meaningful than those generated by traditional topic models. For purposes of comparison, we report some theoretical characteristics of five closely related models in Table 2. Although the two methods above have achieved relatively good performance, there are still some challenges: (1) the first method fails to realize sparse representation in real sense, (2) the second method takes few prior information of relevance between sparse coefficients into consideration.

- *Min Peng and Qianqian Xie are with School of Computer Science, WuHan University, WuHan, China. E-mail: pengm@whu.edu.cn, xieq@whu.edu.cn*
- *Hua Wang and Yanchun Zhang are with Centre for Applied Informatics, Victoria University, Melbourne, Australia. E-mail: hua.wang@vu.edu.au, yanchun.zhang@vu.edu.au*
- *Gang Tian is with School of Computer Science, WuHan University, WuHan, China. E-mail: tiang2008@whu.edu.cn*

TABLE 1
Example Topic Words.

topic 1	obama	appl	relationship	facebook	este	enter	hit	high	lie
topic 2	lauder	lip	facebook	enter	president	nail	follow	win	store
topic 3	tweet	estee	instagram	enter	high	may	spread	obama	win
topic 4	obama	enter	high	metallic	stock	lacquer	iphon	facebook	ipad

In this paper, we present a novel approach to learn sparse latent representations efficiently. The approach is based on our recent work of Sparse Topical Coding with Sparse Groups (*STCSG*)[16]. *STCSG* is a non-probabilistic formulation of topic models for discovering latent representations of large collections of data, by introducing sparse prior. Through imposing sparse groups, *STCSG* relaxes the normalization constraint of the inferred representations and can model the sparsity of word, topic and document codes effectively. However, *STCSG* bypasses the prior information of relevance between sparse coefficients for using traditional sparsity-inducing regularization. To tackle the above challenges and flaws of *STCSG*, we present a novel Bayesian hierarchical sparse topical coding, the Bayesian Sparse Topical Coding with Poisson Distribution (*BSTC-P*), which is an essential Bayesian extension of *STCSG* and have greater flexibility in employing the structural information of the sparse solution. It can learn a sparser and more meaningful representation, and exploit the keywords which correlate strongly with one topic, but are weakly related to other topics as far as possible, thus forms excellent ability to express the semantics of topics. *BSTC-P* utilizes Poisson distribution to model discrete word counts, and uses Gamma-Jeffrey distribution, to model the probability distributions of keywords in the vocabulary, probability distributions of topics in the keywords semantic space, and the topic basis. By using hierarchical Laplace prior, it can discover more compact and efficient coding than *STMs* which utilize non-hierarchical sparse prior and traditional sparse-inducing regularization. However, it still fails to find the sparsest optimal solution, due to taking the Gamma distribution as the prior distribution of sparse solution, since it can affect the convergence of *BSTC-P*. Therefore, we propose the sparsity-enhanced *BSTC*, Bayesian Sparse Topical Coding with Normal Distribution (*BSTC-N*), which utilizes Normal distribution to model word count, and incorporates zero mean Normal-Jeffrey prior to model probability distributions of keywords in the vocabulary, probability distributions of topics in the keywords semantic space, and the topic basis. Generally, for *BSTC-N*, it is more likely to obtain the sparsest optimal solution and can achieve sparser and more meaningful latent representations than *BSTC-P*. Noticed that it is unsuitable to model text and discrete data.

The main contributions of this paper are listed as follows:

1) To the best of our knowledge, we are the first one to employ sparse hierarchical prior for sparse topical coding. We design a novel Bayesian hierarchical topic model *BSTC-P*, to obtain more accurate and effective document, topic and word-level sparsity by introducing sparse Bayesian learning. In order to derive the sparsest optimal solution and more compact sparse representations further, we propose a novel Bayesian hierarchical topic model *BSTC-N*.
2) We incorporate the Expectation Maximization algorithm (EM) and Variational Inference to efficiently approximate the posterior of these two models.
3) We evaluate the effectiveness and efficiency of our models by conducting experiments on 20 Newsgroups and Twitter dataset. Experimental results show that these two models outperform other baselines.

TABLE 2
Theoretical Comparison of Some Topic Models.

model	***LDA***	***Dual-ST***	***STC***	***GSTC***	***STCSG***
Document sparsity	-	no	yes	yes	yes
Topic sparsity	-	yes	no	no	yes
Word sparsity	-	no	yes	no	yes
Sparsity control	-	indirectly	directly	directly	directly

2 Related Work

There have been many works on sparse topic model for obtaining sparse latent representations. Our work is related to the following lines of literature.

2.1 Sparsity-Enhanced Probabilistic Topic Models (*Sparsity-PTMs*)

There are many sparsity-enhanced probabilistic topic models which aim at extracting meaningful latent word, topic and document representations, by imposing sparse priors on LDA-based models. Wang et al. [25], [26] presented the focused topic model (FTM) for learning sparse topic mixture patterns via using a sparse binary matrix drawn from an IBP to enforce sparsity in the latent document representations. Based on the FTM, Chen et al. [27] presented cFTM via leveraging contextual information about the author and document venue, in which the hierarchical beta process is employed to infer the focused set of topics associated with each author and venue. Archambeau et al. [28] proposed IBP-LDA, in which the four-parameter IBP compound Dirichlet process (ICDP) is used to account for the very large number of topics present in large text corpora and the power-law distribution of the vocabulary of natural languages. Similar to IBP-LDA, dual-sparseTM [12] was proposed by using the "Spike and Slab" prior to decouple the sparsity and smoothness of the document-topic and topic-word distributions. Doshi-Velez et al. [23] introduced Graph-Sparse LDA by leveraging knowledge of relationships between words, in which topics are summarized by a few latent concept-words from the underlying graph that explains the observed words. Based on LDA, Kujala [29] proposed sparse topic model LDA-CCCP, in

which the Concave-Convex Procedure (CCCP) is utilized to optimize over the LDA objective and produce sparse model. However, all the methods above lack the ability of controlling the posterior sparsity directly.

2.2 Sparsity-Enhanced Non-probabilistic Topic Models (*Sparsity-NPTMs*)

There are also some non-probabilistic sparse topic models [9], [8], [16] which can directly control the sparsity by imposing regularizers such as lasso, group lasso and sparse group lasso. Methods like matrix factorization (e.g.[7], [30], [31]) formalized topic modeling as a problem of minimizing loss function regularized by lasso, group lasso and so on. But, representations learned by these models are usually non-positive. Furthermore, sparse coding was introduced to non-probabilistic topic models. Zhu et al. [9] proposed sparse topical coding (*STC*) by using mixture regularizers, for discovering latent representations of large collections of data. However, *STC* is not able to discover group sparsity patterns. Bai et al. [8] devised group sparse topical coding (*GSTC*) by imposing group sparse, resulting in better performance than *STC*. But they fail to achieve fully sparse of topics per document, terms per topic, and topics per term. Subsequently, Peng et al. [16] presented sparse topical coding with sparse groups (*STCSG*) to find latent word, topic and document representations of texts. Than et al. [14] proposed Fully Sparse Topic Model (*FSTM*), which can quickly learn sparse topics, infer sparse latent representations of documents, and help significantly save memory for storage. Unlike *PTMs*, they succeed in directly realizing sparse posterior representations in a real sense. However, they still fail to take advantage of prior information of relevance between sparse coefficients.

2.3 Sparse Bayesian Learning

Sparse Bayesian Learning (*SBL*) was proposed by Tipping [32] for obtaining sparse solutions to regression and classification tasks. From then on, *SBL* was widely introduced to compressive sensing (CS), such as [33], [36], [35]. Unlike traditional CS which utilizes lasso, these methods have favorable performance, even if there exists strong correlation between columns of sensing matrix. Wipf et al. [33] motivated the *SBL* cost function as a vehicle for finding sparse representations of signals from overcomplete dictionaries. This *SBL* frame retains a desirable property of the lasso diversity measure and is capable of obtaining a sparser solution by using Jeffreys super prior. This is a non-informative prior distribution for a parameter space, and has been widely used in Bayesian analysis [36], [44]. [34], [35] utilized hierarchical form of prior distributions to model the sparsity of the unknown signals that have high degree of sparsity. Chien et al. [17] presented a new Bayesian sparse learning approach to select salient lexical features for sparse topic modeling based on *LDA*. Moreover, Minjung et al. [37] proposed fully Bayesian formulation of lassos. Compared with traditional penalty-based algorithm (such as lasso, Basis Pursuit), Sparse Bayesian Learning has many obvious advantages: 1) The global minimum of L_1-based algorithm is not the true sparsest solution in the absence of noise [38]. Sparse Bayesian Learning, however, is just the opposite case and is a better alternative [33]; 2) It has been proved that Sparse Bayesian Learning is equivalent to an iterative reweighted L_1 minimization [39]. Meanwhile, it has been pointed out that the true sparsest solution can be easily induced by using the iterative reweighted L_1 minimization [40].

In our work, we propose *BSTC-P* by employing *SBL* to obtain a sparser and more meaningful representation. On one hand, it can directly control the posterior sparsity like *sparse-NPTMs*. On the other hand, it can infer sparse document, topic and word proportions like *sparse-PTMs*. Moreover, we employ the zero mean Normal-Jeffreys hierarchical prior via mathematic approximation to achieve the sparsest optimal solutions in *BSTC-N*.

3 Bayesian sparse topical coding with Poisson distribution

Firstly, we define that $D = \{1, ..., M\}$ is a document set with size M, $T = \{1, ..., K\}$ is a topic collection with K topics, $V = \{1, .., N\}$ is a vocabulary with N words, and $w_d = \{w_{d1}, .., w_{d|I|}\}$ is a vector of terms representing a document d, where I is the index of words in document d, and $w_{dn}(n \in I)$ is the frequency of word n in document d. Moreover, we denote KS_d as the keyword set of document d, and $\beta \in \mathbb{R}^{K \times N}$ as a dictionary with k bases. Additionally, all the notations used in this paper are summarized in Table 3.

TABLE 3
Variables and Notations.

Notation	Meaning
D	document set
T	topic collection
V	vocabulary
$w_{d,.}$	word vector of document
$S_{d,.}$	document code of document d
$s_{d,n}$	word code of n in document d
KS_d	keyword set of document d
β	topic dictionary d
$\theta_{d,.}$	topic representation d
$\phi_{d,.t}$	keyword proportion of t
IKS	increased keywords set
$b^0_{d,.t}, b^1_{d,k}, b^2_k, c^1_{d,.k}$	parameters of gamma distribution
$\sigma^2_{d0,n}, \sigma^2_{d1,.t}, \sigma^2_{d2,k}, \sigma^2_{3,k.}$	parameters of normal distribution

Definition 1 (Word Code, Document Code, Topic Presentation). In this work, a word code $s_{d,n}(n \in I)$ in a document d is defined as the empirical word-topic assignment distribution $p(z(n) = k)$, where $z(n)$ is the topic of word n. A document code $S_{d,.}(d \in D)$ can be regarded as an admixture distribution over words in the document d. The topic presentation θ_d of a document d is an admixture proportion on topic k.

Definition 2 (Keyword, Keyword proportion). In this paper, a keyword of a document d is the word that correlates strongly with one topic, but is weakly related to other topics. A keyword proportion $\phi_{d,.t}(t \in ks_d)$ is an admixture distribution over words in document d.

3.1 Probabilistic Generative Process for *BSTC-P*

Similar to *STCSG*, *BSTC-P* assumes that each word count is a latent variable, and can be reconstructed by linear combination of each keyword proportion and each keyword code with the counterpart column in topic dictionary. To better understand the above assumption, we provide a graphic describing in Figure 1 . Unlike *STCSG*, *BSTC-P*

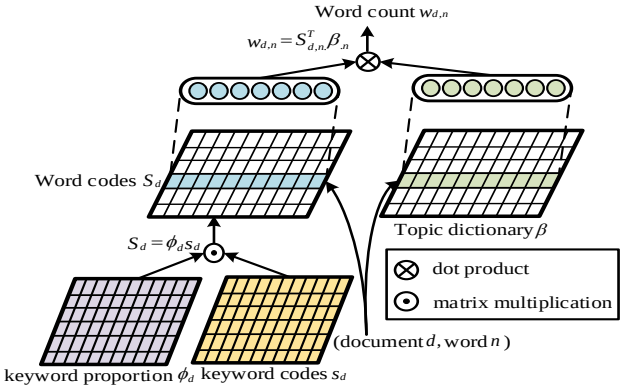


Fig. 1: A graphic describing the reconstruction assumption.

introduces the keyword set and imposes the sparse prior on each dictionary to improve the learning effect of sparse representations. It aims at discovering accurate and meaningful document, topic and word sparse representations for large scale of short texts. According to *STCSG* and previous works, it can be easily inferred that the document codes are the average aggregation of the word code vectors, and topic codes can be also represented by word code vectors with the dictionary. In summary, *BSTC-P* is a Bayesian hierarchical latent variable model and Bayesian extension of *STCSG* by exerting Sparse Bayesian Learning (hierarchical sparse prior). For simplification purposes, we suppose that the observed word counts are independent to each other. The probabilistic generative process of *BSTC-P* is presented as follows (*BSTC-P* is depicted in Figure 2 graphically):
For each topic $k \in \{1, ..., K\}$:

Sample the topic dictionary vectors: $\beta_{k.} \sim Gamma(1, b_{k.}^2)$. For each document $D = \{1, ..., M\}$:

1) For each keyword $t \in KS_d$:
   Sample the keyword proportion: $\phi_{d,.t} \sim Gamma(1, b_{d,.t}^0)$.
2) For each topic $k \in \{1, ..., K\}$:
   Sample the keyword code vectors: $s_{d,.k} \sim Gamma(1, b_{d,k}^1\, c_{d,.k}^1)$.
3) For each observed word $n \in I$:
   Sample the latent word count: $w_{d,n} \sim p(w_{d,n} \mid (\phi_{d,n.}\, s_{d,.})^T \beta_{.n})$.

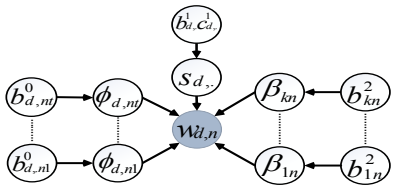


Fig. 2: The graphical model of *BSTC-P*.

In this process, there are several assumptions:

1) Each word count is generated by a Poisson distribution: $Poiss(w_{d,n}; (\phi_{d,n.}\ s_{d,.})^T \beta_{.n})$. The reason for choosing Poisson distribution is the same as in [44], [45], [11].
2) Each row of topic dictionary is generated from a Gamma distribution: $Gamma(\beta_{k.}; 1, b_{k.}^2)$, where $b_{k.}^2 = \{b_{k1}^2, b_{k2}^2 ..., b_{k|I|}^2\}$ is a parameter of Gamma distribution.
3) Each column of keyword codes in document d is derived from Gamma distribution: $Gamma(s_{d,.k}; 1, b_{d,k}^1\, c_{d,.k}^1)$, where $b_{d,k}^1$ and $c_{d,.k}^1 = \{c_{d,1k}^1, c_{d,2k}^1, ..., c_{d,tk}^1\}$ are the parameters of Gamma distribution.
4) Each column of keyword proportion in document d is also derived from Gamma distribution: $Gamma(\phi_{d,.t}; 1, b_{d,.t}^0)$, where $b_{d,.t}^0 = \{b_{d,1t}^0, b_{d,2t}^0, ..., b_{d,|I|t}^0\}$ is the parameter of Gamma distribution.

3.2 Hierarchical Prior Structure of *BSTC-P*

The observation noise of this model is independent and followed in Poisson distribution [18], that is:

$$p(w_{d,n} \mid \phi_{d,n.}, s_{d,.}, \beta_{.n}) = Poisson(w_{d,n}; (\phi_{d,n.}\, s_{d,.})^T \beta_{.n}), \quad \phi_{d,n.} \geq 0, s_{d,.} \geq 0, \beta_{.n} \geq 0. \tag{1}$$

Unlike using the whole word set, we consider extracting the keyword set to represent the semantic of each topic. For keyword proportions, a Gamma distribution is employed to model it, then we deduce

$$p(\phi_{d,.t} \mid 1, b_{d,.t}^0) = b_{d,.t}^0\, e^{-\phi_{d,.t}/b_{d,.t}^0}. \tag{2}$$

Since the Gamma distribution is the conjugate prior of Poisson distribution, it can be generally chosen as the prior for the Poisson distribution. In the first stage of hierarchical prior, we employ the following prior on $s_{d,.k}$ to achieve sparse codes at document and topic-level:

$$p(s_{d,.k} \mid 1, b_{d,k}^1\, c_{d,.k}^1) = Gamma(s_{d,.k}; 1, b_{d,k}^1\, c_{d,.k}^1) \tag{3}$$

In the second stage, we exert the non-informative Jeffreys super prior on each $b_{d,.t}^0$, $b_{d,k}^1$ and $c_{d,.k}^1$ independently as follows:

$$p(b_{d,.t}^0) \propto \frac{1}{b_{d,.t}^0}, p(b_{d,k}^1) \propto \frac{1}{b_{d,k}^1}, p(c_{d,.k}^1) \propto \frac{1}{c_{d,.k}^1}. \tag{4}$$

The Jeffreys prior is an improper prior, and has been proved that it can enforce sparsity in classification, regression models and compressive sensing. From the two stages above, we have

$$p(\phi_{d,.t} \mid 1, b_{d,.t}^0) = -\prod_i b_{d,it}^0\, e^{-\phi_{d,it}/b_{d,it}^0}, \tag{5}$$

$$p(s_{d,.k} \mid 1, b_{d,k}^1\, c_{d,.k}^1) = -\, b_{d,k}^{1\,(|\,KS_d\,|)} \prod_i c_{d,ik}^1\, e^{-\,s_{d,ik}\,/\,b_{d,k}^1\, c_{d,ik}^1}. \tag{6}$$

Since $\phi_{d,.t} \geq 0, s_{d,.k} \geq 0$, Eq. (5) and (6) can be rewritten as:

$$p(\phi_{d,.t} \mid 1, b_{d,.t}^0) = -\prod_i b_{d,it}^0\, e^{-|\phi_{d,it}|/b_{d,it}^0}, \tag{7}$$

$$p(s_{d,.k}\,|1,b^1_k\,c^1_{d,.k}) = -\,b^{1^{(|\,KS_d\,|)}}_k \prod_i c^1_{d,ik}\,e^{-\,|s_{d,.k}\,|/\,b^1_{d,k}\,c^1_{d,ik}}. \tag{8}$$

In Eq. (7) and (8), we can find that the hierarchical prior on $\phi_{d,.t}$ is approximately equivalent to a hierarchical Laplace prior, and the hierarchical prior on $s_{d,.k}$ is approximately equivalent to a hierarchical Multi-Laplace prior. The hierarchical Laplace and Multi-Laplace prior have been proved to be superior to lasso and group lasso in variable selection, for employing prior information of relevance between sparse coefficients. In this paper, we consider each column of s as a group. We also impose hierarchical prior for each $\beta_{k.}$ the same as $\phi_{d,.t}$, following the future work of *STCs*. The two-layer hierarchical prior structure of $\beta_{k.}$ can be expressed as:

$$p(\beta_{k.}\,|1,b^2_{k.}) = b^2_{k.}\,e^{-\,\beta_{k.}/b^2_{k.}}, p(b^2_{k.}) \propto \frac{1}{b^2_{k.}}, \tag{9}$$

According to Eq. (9), we infer

$$p(\beta_{k.}\,|1,b^2_{k.}) = -\prod_i b^2_{ki}\,e^{-\,\beta_{ki}/b^2_{ki}}, \tag{10}$$

Similar to $\phi_{d,.t}$, Eq. (10) can be rewritten as

$$p(\beta_{k.}\,|1,b^2_{ki}) = -\prod_i b^2_{ki}\,e^{-\,|\beta_{ki}\,|/b^2_{ki}}, \tag{11}$$

From Eq. (11), we can see that the two-layer hierarchical prior of $\beta_{k.}$ is a hierarchical Laplace prior. In summary, *BSTC-P* is a four-stage hierarchical model. The first stage is to reconstruct latent observed word counts by Poisson distribution. The second part is to generate keyword proportions for extracting focused words. The third and fourth result in parameter-free group Laplace prior by hierarchical prior, for extracting focused topics. Compared with *STCs*, *BSTC-P* makes the exploiting of the structure information more flexible for taking parameterized Gamma distribution as the prior distribution of sparse solution, and has the advantages of obtaining sparse word, topic and document representations by bringing *SBL* framework.

3.3 Bayesian Inference of *BSTC-P*

3.3.1 Bayesian Inference

Now, we aim to estimate unknown parameters which enable posterior maximum at the same time. According to Bayes' Rule, the posterior distribution of this model is

$$\begin{aligned} &p(\phi,s,\beta|w)\\ &\propto p(w|\phi,s,\beta)p(\phi|\,b^0)p(s|\,b^1,c^1)p(\beta|\,b^2)p(b^0)p(b^1)p(c^1)p(b^2). \end{aligned} \tag{12}$$

Then, for each document d, the logarithm of Eq. (12) is:

$$\begin{aligned} &\ln(p(\phi_{d,.}\,,s_{d,.},\beta_{.}\,|\,w_{d,.})\\ &=\sum_d\Bigg(w_{d,.}\ln\sum_t(\phi_{d,.t}\,s_{d,t.})^T\,\beta_{.}\\ &\quad-\sum_t((\phi_{d,.t}\,s_{d,t.})^T\,\beta_{.}-\phi_{d,.t}/b^0_{d,.t})\\ &\quad+\sum_k(-s_{d,.k}/b^1_{d,k}\,c^1_{d,.k}-\beta_{.k}/b^2_{.k})\Bigg). \end{aligned} \tag{13}$$

By inferring the objective function (13), we can learn keyword proportions, word codes, as well as the dictionaries. The Variational Inference [45], [46] is widely used for inferring of various complex models in Bayesian estimation and machine learning. According to the Variational Bayes (VB) method, the lower bound of the marginal log-likelihood is tight for the exact posterior $q = q(\phi,s,\beta)$, that is:

$$\begin{aligned} &L_w(\phi,s,\beta,b^0,b^1,c^1,b^1)\\ &\geq\ \langle\log p(w,\phi,s,\beta|\,b^0,b^1,c^1,b^1)\rangle_q + H[q]\\ &=\ ELBO_{VB}[q], \end{aligned} \tag{14}$$

where $q = q(\phi,s,\beta)$ is an instrumental distribution and $H[q]$ is its entropy. Therefore, the expectation of the exact posterior is approximately equal to

$$\begin{aligned} &Q\,(\phi,s,\beta|w)\\ &\propto\sum_d(\sum_t(w_{d,.}\ln(\phi_{d,.t}\,s_{d,t.})^T\,\beta_{.}-(\phi_{d,.t}\,s_{d,t.})^T\,\beta_{.}\\ &\quad-\phi_{d,.t}/b^0_{d,.t}-\ln b^0_{d,.t})+\sum_k(-s_{d,.k}/b^1_{d,k}\,c^1_{d,.k}-\beta_{.k}/b^2_{.k})), \end{aligned} \tag{15}$$

Jensens Inequality plays a central role in the derivation of the Expectation Maximization algorithm [48] and Variational Inference [45] to facilitate the calculation. In Eq. (13), we employ the Jensens inequality again for ease of calculation. According to Jensens inequality, we maximize the variational objective with respect to the variational parameters (ϕ,s,β) of q. For each keyword $t \in KS_d$ in document d, we deduce,

$$\begin{aligned} &q(\phi_{d,.t})\propto Gamma(\phi_{d,.t};1,b^0_{d,.t})\\ &\quad b^0_{d,.t}\equiv\left(\frac{1+\sum\limits_n w_{d,n}}{(b^0_{d,.t})^0}+(s_{d,t.}\,\beta_{.})^T\right)^{-1}\\ &\quad\phi_{d,.t}=\langle b^0_{d,.t}\rangle. \end{aligned} \tag{16}$$

For each topic $k \in \{1,...,K\}$ in document, we deduce,

$$\begin{aligned} &q(s_{d,.k})\propto Gamma(s_{d,.k};1,b^1_{d,k}\,c^1_{d,.k})\\ &\quad b^1_{d,k}\equiv\left(\frac{1+\sum\limits_n(w_{d,n}-c_{1,n})}{(b^1_{d,k})^0}+\sum_t(\beta_{k.}\,\phi_{d,.})^T\right)^{-1}\\ &\quad c^1_{d,.k}\equiv\left(\frac{1+\sum\limits_n(w_{d,n}-c_{1,n})}{(c^1_{d,.t})^0}+(\beta_{k.}\,\phi_{d,.})^T\right)^{-1}\\ &\quad s_{d,.k}=\langle b^1_{d,k}\,c^1_{d,.k}\rangle. \end{aligned} \tag{17}$$

After we have inferred the latent representations (ϕ,s) of all documents, we update the dictionary as follow:

$$\begin{aligned} &q(\beta_{k.})\propto Gamma(\beta_{k.};1,b^2_{k.})\\ &\quad b^2_{k.}\equiv\left(\frac{1+\sum\limits_d\sum\limits_n(w_{d,n}-c_{1,dn})}{(b^2_{k.})^0}+\sum_d(\phi_{d,.}\,s_{d,.k})^T\right)^{-1}\\ &\quad\beta_{k.}=\langle b^2_{k.}\rangle, \end{aligned} \tag{18}$$

where $c_1=\sum\limits_d\sum\limits_{i=1\wedge i\neq k}^{K}\phi_{d,.}\,s_{d,.i}\,\beta_{i.}$ and $c_2=c_1$. These three equations (Eq. (16), Eq. (17) and Eq. (18) are analogous to the posterior of EM in E step and they all follow the Gamma distribution.

3.3.2 Dynamic Update of keyword Set

The quality of the keywords is important to construct meaningful word, topic and document representations. However,

in practice, the number of keywords should neither be too large nor too small. Generally, it is hard to find the optimal number of keywords. What's worse, the initialized keywords are usually not very accurate. Therefore, there is still a problem of how to learn reliable keyword, even though we have inferred their clear update expressions. To solve this problem, we utilize a robust iterative approach [49], which can improve the quality of keywords, and determine the number of keywords, thus can approximate more stable topic representations. We initialize keywords by TFIDF score. As shown in Figure 3, after generating ϕ, s, β, the prediction information can be regarded as feedback to word space. To reselect keywords, we recalculate TFIDF score of each word according to the learned ϕ, s, β. That is:

$$w^* = (\phi s)^T \beta, \tag{19}$$

where w^* is the weight of words. Then, we choose the words that are highly relevant to the topic as the keywords, namely,

$$KS = KS \cup IKS, \tag{20}$$

where IKS is the increased keyword sets. To sum up, the learning algorithm can be summarized in Algorithm 1.

Algorithm 1 Bayesian Inference for *BSTC-P*

Input: ϕ, s, β, D, KS;
Output: ϕ, s, β;
repeat
  for each $d \in \{1, ..., M\}$ **do**
    for each keyword $t \in KS_d$ **do**
      $E_{\phi_{d,.t}} = \langle b^0_{d,.t} \rangle$;
      caculate $b^0_{d,.t}$ according to Eq. (16);
    end for
    for each topic $k \in \{1, ..., K\}$ **do**
      $E_{s_{d,.k}} = \langle b^1_{d,k} \, c^1_{d,.k} \rangle$;
      caculate $b^1_{d,k}, c^1_{d,.k}$ according to Eq. (17);
    end for
  end for
  $E_{\beta_{k.}} = \langle b^2_{k.} \rangle$;
  caculate $b^2_{k.}$ according to Eq. (18);
  $w^*_{d,.} = (\phi_{d,.} \, s_{d,.})^T \beta_{.}$;
  $KS = KS \cup IKS$;
until $KS - (KS \cup IKS) = NULL$

4 Bayesian sparse topical coding with Normal distribution

In the above sections, *BSTC-P* is used to learn meaningful latent sparse representations. Although *BSTC-P* performs better than *STCSG* and other *STMs*, it does not succeed in obtaining the sparsest optimal solution for the stability problems from adopting the Gamma distribution as the prior distribution of sparse solution. This results in the proper prior for sparse coefficients, therefore, cannot ensure the global convergence of model. To address the weakness of *BSTC-P*, we next propose the *BSTC-N* via mathematic approximation. In this model, unlike *BSTC-P*, we expect to construct the improper sparse prior for sparse coefficient, so as to promote the sparse solution and ensure the convergence of model.

4.1 Probabilistic Generative Process for *BSTC-N*

Similar to *STCSG* and *BSTC-P*, in this model, word count is also treated as a latent variable, and can be reconstructed by the linear combination of each keyword proportion, each keyword code with the counterpart column in topic dictionary. However, in *BSTC-N*, the continuous Normal distribution is used to model the latent word counts rather than Poisson distribution in *BSTC-P*, meanwhile the keyword proportions, word codes and topic dictionary are all subject to Normal distribution with zero mean but not the Gamma distribution in *BSTC-P*. It devotes to discovering accurate and meaningful full sparse representations for large scale of short texts as well. Furthermore, it can acquire the sparsest optimal solution and outperforms *BSTC-P* in finding sparse coding for using zero mean Normal-Jeffrey-based hierarchical prior. To derive this model, we first present the generative process of *BSTC-N* as follows (The whole process also can be seen in Figure 4).

For each topic $k \in \{1, ..., K\}$:
  Sample the topic dictionary vectors: $\beta_{k.} \sim N(0, \sigma^2_{3,k.})$.
  For each document $D = \{1, ..., M\}$:

1) For each keyword $t \in KS_d$:
   Sample the keyword proportion: $\phi_{d,.t} \sim N(0, \sigma^2_{d1,.t})$.
2) For each topic $k \in \{1, ..., K\}$:
   Sample the keyword code vectors: $s_{d,.k} \sim N(0, \sigma^2_{d2,k})$.
3) For each observed word $n \in I$:
   Sample the latent word count: $w_{d,n} \sim p(w_{d,n} \mid (\phi_{d,n.} \, s_{d,.})^T \beta_{.n}, \sigma^2_{d0,n})$.

Several simplified assumptions are made in this model:

1) As the limit of Poisson distribution is Normal distribution, we therefore assume the word count is generated by Normal distribution $N((\phi_{d,n.} \, s_{d,.})^T \beta_{.n}, \sigma^2_{d0,n})$, where $\sigma^2_{d0,n}$ is the appropriate covariance.
2) For the convenience of calculations, we assume the topic dictionary is the Normal distribution with zero mean i.e., $N(0, \sigma^2_{3,k.})$, where $\sigma^2_{d3,k.}$ is the appropriate covariances.
3) For similar reason, the keyword code vector and each column of keyword proportion are also Normal distributions with zero mean i.e., $N(0, \sigma^2_{d2,.k})$ and $N(0, \sigma^2_{d1,.t})$, where $\sigma^2_{d2,k}$ and $\sigma^2_{d1,.t}$ are appropriate covariances.

4.2 Hierarchical Prior Structure of *BSTC-N*

In this paper, we formulate *BSTC-N* as a Bayesian problem. According to the above generation process, the reconstruction error is

$$\begin{aligned} &p\left(w_{d,n} \mid \phi_{d,n}, s_{d,.}, \beta_{.n}\right) \\ &\quad = N(0, (\phi_{d,n.} \, s_{d,.})^T \beta_{.n}), \phi_{d,n} \geq 0, s_{d,.} \geq 0\,, \beta_{.n} \geq 0. \end{aligned} \tag{21}$$

To achieve sparse word representations, we expect to discover keywords to represent semantics of the topic well, while discarding the meaningless words. We also expect that only a little set of topics are non-zeros and a little set of words are non-zeros, so as to achieve document and topic

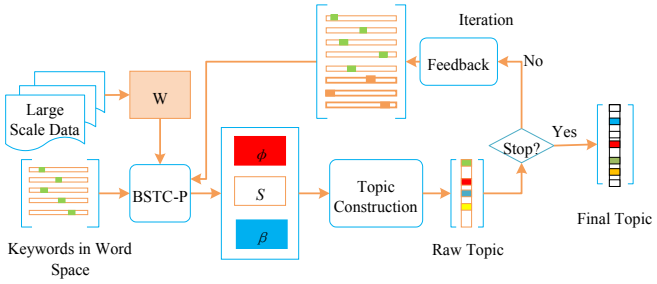


Fig. 3: Procedure of the iterative approach.

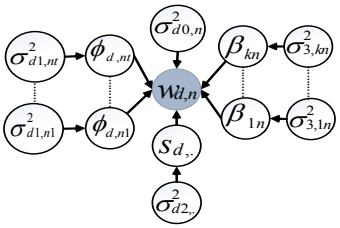


Fig. 4: The graphical model of *BSTC-N*.

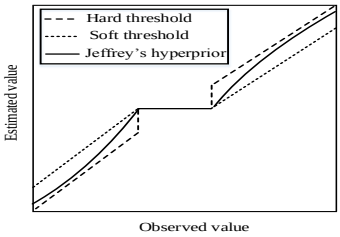


Fig. 5: The estimation rule.

sparse representations. They could be achieved by exerting hierarchical prior on keyword proportions, keyword codes and the topic basis. A keyword proportion is generated by

$$p(\phi_{d,.t}\,|0,\sigma^2_{d1,.t}) = \frac{1}{\sqrt{2\pi}\,\sigma^2_{d1,.t}} \exp^{-\frac{\phi^2_{d,.t}}{2\,\sigma^2_{d1,.t}}}. \tag{22}$$

In the second step, we employ the non-informative Jeffreys super prior on parameter $\sigma^2_{d1,.t}$ like *BSTC-P*, that is:

$$p(\sigma^2_{d1,.t}) \propto \frac{1}{\sigma^2_{d1,.t}}. \tag{23}$$

Similarly, keyword codes and topic basis are also generated by that hierarchical prior as follows:

$$p(s_{d,.k}\,|0,\sigma^2_{d2,k}) = \frac{1}{\sqrt{2\pi}\,\sigma^2_{d2,k}} \exp^{-\frac{{s_{d,.k}}^2}{2\,\sigma^2_{d2,k}}}, p(\sigma^2_{d2,k}) \propto \frac{1}{\sigma^2_{d2,k}}, \tag{24}$$

$$p(\beta_{k.}\,|0,\sigma^2_{3,k.}) = \frac{1}{\sqrt{2\pi}\,\sigma^2_{3,k.}} \exp^{-\frac{{\beta_{k.}}^2}{2\,\sigma^2_{3,k.}}}, p(\sigma^2_{3,k.}) \propto \frac{1}{\sigma^2_{3,k.}}, \tag{25}$$

The hierarchical prior that is constructed by Normal distribution with zero mean and Jeffrey's super prior, also has better capacity to select variables for utilizing prior information of relevance between sparse coefficients. Furthermore, compared with lasso and hierarchical Laplacian prior, this method tends to obtain the sparsest solution because of the improper hierarchical prior to sparse coefficients [32], [33]. Figure 6 shows the contour plots of the penalty functions in two dimensions resulting from Laplace prior and Jeffrey's prior models. In Figure 6, the fit terms are elliptical contours, and the solution is the first place that this contour touches the penalty contour[54]. From Figure 6, we can see the penalty contour of Jeffrey's prior has more chance to touch the corner, that is to say, the penalty function from Jeffrey's prior has more chance to lead a sparse solution than Laplace prior [52]. Figure 7 displays soft-thresholding

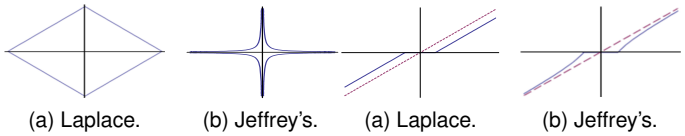


Fig. 6: Contour.

Fig. 7: Soft-thresholding.

plots for Laplace and Jeffreys prior. Soft-thresholding plots show the sparsity properties of regularizers and the solution of the penalized problem [52], [53]. From Figure 7, we can find that the Jeffrey's prior has a smooth penalty with non-zero coefficients asymptotically approaching the solution without shrinkage. As shown above, we can observe that the Jeffrey's prior results in better performance than the Laplace prior.

We also note that there are sparsity regularizations which encourage more sparsity than the hierarchical Gaussian-Jeffreys prior, such as hard-thresholding. However, we choose this prior for following reasons: 1) The estimation rule produced by the EM Algorithm with the Jeffreys hyperprior combines the advantages of hard and soft threshold estimation rules. For comparison purposes, Figure 5 plot the estimation rule produced by the EM Algorithm with the Jeffreys hyperprior, alongside the well-known hard and soft threshold estimation rules. It shows that the Jeffreys hyperprior is close to the soft rule at small value, thus effectively behaving like a shrinkage rule, and it approaches the hard rule at large value, avoiding the undesirable bias incurred with the soft rule [36]. In conclusion, the estimation rule of the Jeffreys hyperprior places itself between two threshold estimation rules, combining the advantages of both. 2) Although the Jeffreys prior has a non-convex contour, its update rule of the expectation-maximization (EM) algorithm suits the case when a closed form solution can be derived due to its hierarchical structural with normal distribution. Because the complete log-posterior is easy to deduce when we regard the hyper-parameter as a hidden variable [36]. 3) With the Jeffreys prior, the tuning or adaptive estimation of the parameters of the prior is unnecessary.

In summary, *BSTC-N* also has four stages, the first stage is to reconstruct latent observed word counts by Normal distribution. The second part is to generate keyword proportions, for word sparsity. The third and fourth result in parameter-free group sparse prior by hierarchical prior, for topic and document sparsity. Both *BSTC-P* and *BSTC-N* are Bayesian hierarchical latent variable models based on the same core idea, while the only difference among them is the priors for latent variables. Furthermore, we provide a comparison between them: 1) *BSTC-P* is much better suited to model text data than *BSTC-N*. In *BSTC-P*, the observed word count is sampled from the Poisson distribution, which is generated from the Normal distribution in *BSTC-N*. Many real data can be well expressed via Gaussian noise model. However, this is an inappropriate assumption for text data, which is often represented as the word count vector. For the discrete word count, the discrete Poisson distribution

is a better choice than the continuous Gaussian distribution. 2) Compared to *BSTC-N*, *BSTC-P* is more natural to constrain the feasible domains (non-negative for modeling word counts). Yet the Normal distribution based *BSTC-N* could take fractional or negative values, which may lead to inefficient learning. 3) From the perspective of sparsity inducing, *BSTC-N* is more possible to find the sparsest optimal solution, due to the Normal-Jeffrey's hyper prior.

4.3 The EM algorithm of *BSTC-N*

We aim to infer the keyword proportions, word codes and dictionaries by maximizing marginal likelihood directly and efficiently. According to the generative process, we have the posterior distribution of *BSTC-N* as follow:

$$p(\phi, s, \beta|w) \propto p(w|\phi, s, \beta, \sigma_0^2)p(\phi|\,\sigma_1^2)p(s|\,\sigma_2^2)p(\beta|\,\sigma_3^2) \\ p(\sigma_0^2)p(\sigma_1^2)p(\sigma_2^2)p(\sigma_3^2), \tag{26}$$

For each document d after some mathematical transformations, we can obtain the logarithm of Eq. (26) as follow:

$$\begin{aligned} &\log p(\phi_d\,, s_d, \beta_d\,|\,w_d) \\ &\propto \sum_d ((\sum_n -\log \sigma_{d,0n}^2 - \frac{||\,w_{d,n} - (\phi_{d,n.}\,s_{d,.})^T\,\beta_{.n}\,|^2}{\sigma_{d,0n}^2}) \\ &\quad + \sum_t (-\,\phi_{d,.t}^T\,\psi(\sigma_{d1,.t}^2)\phi_{d,.t}) \\ &\quad + \sum_k (-\,s_{d,.k}^T\,\psi(\sigma_{d2,k}^2)\,s_{d,.k} - \beta_{k.}^T\,\psi(\sigma_{3,k.}^2)\,\beta_{k.})), \end{aligned} \tag{27}$$

where $\psi(\sigma_{d1,.t}^2) = diag({\sigma_{d1,1t}^2}^{-1}, {\sigma_{d1,2t}^2}^{-1}, ..., {\sigma_{d1,|I|t}^2}^{-1})$, $\psi(\sigma_{d2,k}^2) = diag({\sigma_{d2,k}^2}^{-1}, {\sigma_{d2,k}^2}^{-1}, ..., {\sigma_{d2,k}^2}^{-1})$, and $\psi(\sigma_{3,k.}^2) = diag({\sigma_{3,k1}^2}^{-1}, {\sigma_{3,k2}^2}^{-1}, ..., {\sigma_{3,k|\,I\,|}^2}^{-1})$. In the E step, given the current parameter estimates and the observed data, the expected value of the complete log-posterior $Q(\phi, s, \beta|w) \propto \log p(\phi_{d,.}\,, s_{d,.}, \beta_{d,.}\,|\,w_{d,.})$. It is easy to maximize ϕ, s and β, because

$$\begin{aligned} E\,({\sigma_{d1,nt}^2}^{-1}\,|\,\phi_{d,nt}) &= \frac{1}{|\,\phi_{d,nt}\,|^2} \\ E\,({\sigma_{3,kn}^2}^{-1}\,|\,\beta_{kn}) &= \frac{1}{|\beta_{kn}|^2} \\ E\,({\sigma_{d2,k}^2}^{-1}\,|\,s_{d,.k}) &= \frac{1}{||s_{d,.k}\,||^2}. \end{aligned} \tag{28}$$

Therefore, it is easy to maximize Q with respect to ϕ, s and β, yielding

$$\sigma_{d0,.}^2 = \sum_d ||\,w_{d,.} - (\phi_{d,.}\,s_{d,.})^T\,\beta_{.}\,||^2. \tag{29}$$

For each word n in document d, we yield

$$\begin{aligned} \phi_{d,n.} = \sum_d \sum_n ((&s_{d,.}^T\,\beta_{.n}\,\beta_{.n}\,s_{d,.}^T \\ &+ \sigma_{d,0n}^2\,diag\{{\phi_{d,n1}^2}^{-2}, {\phi_{d,n2}^2}^{-2}, ..., {\phi_{d,nt}^2}^{-2}\})^{-1} \\ &s_{d,.}^T\,\beta_{.n}\,w_{d,n}), \end{aligned} \tag{30}$$

As for each topic k in document d, we deduce

$$\begin{aligned} s_{d,.k} = \sum_d (\sum_n (&\frac{\phi_{d,n.}^T\,\beta_{kn}\,\beta_{kn}\,\phi_{d,n.}^T}{\sigma_{d,0n}^2} \\ &+ diag\{\frac{1}{||s_{d,.k}\,||^2}, \frac{1}{||s_{d,.k}\,||^2}, ..., \frac{1}{||s_{d,.k}\,||^2}\})^{-1} \\ &\sum_n \frac{\phi_{d,n.}^T\,\beta_{kn}(w_{d,n} - C_{d,nk})}{\sigma_{d,0n}^2}), \end{aligned} \tag{31}$$

where $C_{d,.k} = \sum_d \sum_{i \neq k} ((\phi_{d,.}\,s_{d,.i})^T\,\beta_{i.})$.

For all document, we can update the dictionary by minimizing the following object function:

$$\begin{aligned} \beta_{.n} = \sum_n ((&\sum_d \phi_{d,n.}\,s_{d,.}^T\,s_{d,.}^T\,\phi_{d,n.} \\ &+ \sigma_{d,0n}^2\,diag({\beta_{1n}^2}^{-2}, {\beta_{2n}^2}^{-2}, ..., {\beta_{Kn}^2}^{-2}))^{-1} \\ &\sum_d \phi_{d,n.}\,s_{d,.}^T\,w_{d,n}). \end{aligned} \tag{32}$$

In this paper, an similar iterative approach is also devised to obtain a preferable keyword set like *BSTC-P*. The learning algorithm can be summarized in Algorithm 2.

Algorithm 2 EM algorithm for *BSTC-N*

Input: ϕ, s, β, D, KS;
Output: ϕ, s, β;
repeat
  for each $d \in \{1, ..., M\}$ **do**
    $\sigma_{d0,.}^2 = \sum_d ||\,w_{d,.} - (\phi_{d,.}\,s_{d,.})^T\,\beta_{.}\,||^2$;
    for each word $n \in \{1, ..., |I|\}$ **do**
      caculate $\phi_{d,n.}$ according to Eq. (30);
    end for
    for each topic $k \in \{1, ..., K\}$ **do**
      caculate $s_{d,.k}$ according to Eq. (31);
    end for
  end for
  caculate $\beta_{.n}$ according to Eq. (32);
  $w_{d,.}^* = (\phi_{d,.}\,s_{d,.})^T\,\beta_{.}$;
  $KS = KS \cup IKS$;
until $KS - (KS \cup IKS) = NULL$

5 Experiments

In this section, we will display the dataset, experimental settings, and evaluation results.

5.1 Dataset and Experimental Setting

We perform experiments on 20 Newsgroups [1] and Twitter dataset to test the performance of *BSTC-P* and *BSTC-N*. The 20 Newsgroups dataset is comprised of 18775 newsgroup articles with 20 categories, and contains 60,698 unique words. The Twitter feeds, which covers 2,068,721 tweets with 10 categories, are gathered by our web crawler [2]. We build up a vocabulary that contains 3000 frequency terms after removing stop words and infrequent words. *BSTC-P* and

1. http://qwone.com/jason/20Newsgroups/
2. http://sc.whu.edu.cn/

BSTC-N are implemented with MATLAB under a desktop with 2.33GHZ Intel processor, Xeon CPU and 8GB RAM. We initialize the keywords proportion, the word code and the topic dictionary to be Gamma distribution in *BSTC-P*. Meanwhile, we initialize the keywords according to their TFIDF score for each document, set $b^0_{d,nt} = 0.05|\, I_d \,|$, $c^1_{d,tk} = 0.05\,|T_d\,|$, $b^1_{d,k} = ||\, c^1_{d,.k}\,||_2$, $b^2_{kn} = 1/k$, $\sigma^2_{d,0} = \frac{1}{N}$, and use the alternate iteration method to solve to initialize the keywords proportion, the word code and the topic dictionary in *BSTC-N* by solving $w_{d,n} \sim (\phi_{d,n.}\, s_{d,.})^T\, \beta_{.n} + \varepsilon$. For 20 Newsgroups, we set the initial number of keywords T to 10, while for Twitter dataset, we set T to 3, and the size of the IKS is $\lfloor T/2 \rfloor$ each time. In experiments, we compare the performance of our two *BSTCs* with the following models:

- *DsparseTM*. *DsparseTM* is a recently proposed dual-sparse topic model that addresses the sparsity in both the topic mixtures and the word usage.
- Sparse Topical Coding with Sparse Groups (*STCSG*). *STCSG* is a fully sparsity-enhanced non-probabilistic topic model which aims at learning word, topic and document sparse representations.
- Sparse Topical Coding (*STC*). *STC* [3] is a sparsity-enhanced non-probabilistic topic model. It has been proven to achieve word and document sparse representations, and perform better than some of the existing models.
- *LDA*. *LDA* [4] is a classical probabilistic topic model, which can induce sparsity as the Dirichlet prior approaches zero, but can't decouple the smoothness.
- *Mixture of Unigrams*. The Mixture of Unigrams assumes that each document is generated by only one topic which generates words independently from the conditional multinomial.

The performance of these methods is evaluated from topic coherence, sparse ratio of latent representations of document, topic and word and classification accuracy of documents.

5.2 Evaluation of Topic Coherence

Topic coherence is a common measure of topic models generalization ability on test data. Newman et al. [55] calculated the pointwise mutual information (PMI) of each word pair to measure the semantic coherence of topics. It has been widely used to measure the statistical independence of observing two words in close proximity [56]. The PMI can be computed as follow :

$$PMI(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)}, \tag{33}$$

where (w_i, w_j) is word pair, $p(w_i, w_j)$ is the joint probability of words w_i and w_j co-occurring in the same document, and $p(w_i)$ is the marginal probability of word w_i appearing in a document. In this paper, we choose top-15 words to calculate the average relatedness of each pair of these words as the PMI score of each topic. Table 4 shows the PMI scores of six candidate methods in two datasets with 120 topics and 50 topics, respectively. From Table 4, we can find that

3. http://bigml.cs.tsinghua.edu.cn/ jun/stc.shtml/
4. http://www.cs.princeton.edu/blei/lda-c/

TABLE 4
Topic Coherence (PMI) of Six Models.

	20NG	**Twitter**
Number of topics	120	50
BSTC-P	1.6698	1.6319
BSTC-N	1.6779	1.6657
DsparseTM	1.6210	1.5510
STC	1.5150	1.3780
LDA	1.3360	1.1989
Mixture of Unigrams	0.6910	1.1210

the proposed *BSTC-P* and *BSTC-N* yield higher PMI scores than *DsparseTM* and *STC* methods in both datasets. This is mainly because *BSTC-P* and BSCT-N utilize the keyword set and hierarchical sparse prior, which achieve more coherent and meaningful topics. It also shows that *BSTC-P* and *BSTC-N* perform well in both long and short documents. As for *BSTC-P* and *BSTC-N*, we can see that *BSTC-P* has a lower PMI score than *BSTC-N* because of the Normal-Jeffrey hierarchical prior of *BSTC-N* has more ability to cover focused topics than Gamma-Jeffrey hierarchical prior of *BSTC-P*. Moreover, we also observe that the four sparse-enhanced topic models *BSTC-P*, *BSTC-N*, *DsparseTM* and *STC* have higher PMI scores than two non-sparse-enhanced topic models *LDA* and Mixture of Unigrams, because their sparsity-induced prior can detect document-topic sparsity.

To further demonstrate the semantics of the topics learned by our models, in Table 5 and 6, we show the top-8 words of learned focused topics of *BSTC-P* in Twitter and 20 Newsgroups. We omit the results of *BSTC-N* in order to save space. It is obvious that the learned topics are clear and meaningful. Such as *color*, *compression*, *graphics*, *photoshop*, *photos* and *polygon* in the topic about graphics, and *bike*, *motorcycle*, *bikes*, *seat*, and *back* in the topic about motorcycles.

TABLE 5
Top Words of Learned Topics in Tweets.

barack obama	**chanel**	**eatee lauder**	**facebook**	**iphone 5s**
obama	coco	estee	facebook	iphon
follow	chanel	lauder	share	ipad
barack obama	love	intens	retweet	photo
win	girl	lip	life	buy
folli	price	high	relationship	apple
romney	chanc	mettalic	twitter	galaxi
presid	style	stock	instagram	phone
vote	pricey	lacquer	davelacki	mini

5.3 Evaluation of Classification accuracy

In this part, we quantitatively investigate the performance of our model in learning meaningful sparse representations in these two datasets. Here, we compare the accuracy of text classification tasks to verify the performance of our model in learning meaningful sparse representations. For 20NG dataset, we use 60% documents as training set and 40% as testing set. As for Twitter dataset, we sample 10% tweets from the Twitter data set, then we use 80% documents for training and 20% for testing. We adopt the

TABLE 6
Top Words of Learned Topics in 20NG.

comp.			misc.	sci.		alt.	soc.
graphics	**ms-windows**	**ibm.pc**	**forsale**	**crypt**	**electronics**	**atheism**	**christian**
color	windows	ibm	trade	encryption	coil	pinko	god
compression	dos	apple	watchy	math	voltage	tribe	church
graphics	screen	pc	market	db	electronics	islamic	jesus
photoshop	load	scsi	trades	signature	connect	authorities	resurrection
photos	program	card	sale	log	wave	muslims	bible
polygon	workplace	video	brand	key	pins	humanitarian	ye
files	unix	floppy	offer	des	wires	ijaz	judah
bmp	keyboard	vram	price	attack	wattage	godless	chronicles

LIBSVM toolbox [5] as the classifier with the sparse document representation of each document. To better understand the behavior of document representations in classification, we sample different ratio of training set (0.2%, 1%, 10% and 100%) as labeled documents. Figure 8 reports the classification accuracy under different sampling ratios of training data. Figure 9 shows the classification accuracy on Twitter dataset. From Figure 8, we can observe that *BSTC-P* and *BSTC-N* outperform other four models all the time due to the hierarchical sparse prior, where the learned document codes are more trenchant among word set. Meanwhile, the sparsity-enhanced models, *BSTC-P*, *BSTC-N*, *DsparseTM* and *STC* perform much better than *LDA* and *M-U* even when the training data is rare. The possible reason is that they focus on obtaining admixture document codes, where keyword features are likely to overfit the classifier to the data, and are immune to rare word co-occurrence information. From Figure 9, it is obvious that *BSTC-P* and *BSTC-N* have higher classification accuracy than basic *STCSG* and other two models (*STC* and *LDA*). One possible reason for the improvement is that, in these two models the learned admixture proportions are more distinct and better than document codes that are learned by other three models. Moreover, in both datasets, *BSTC-N* outperforms the *BSTC-P* mostly. This indicates that the Normal-Jeffrey hierarchical sparse prior is superior to the Gamma-Jeffrey hierarchical sparse prior in learning sparse representations of short texts. Nevertheless, we observed that the accuracy of *BSTC-P* is higher than *BSTC-N* in Figure 8b. It reveals the poor expression of *BSTC-N* in discrete word count, due to the Gaussian distribution assumption. Thus the discrete Poisson distribution method *BSTC-P* yields better performance than *BSTC-N*.

5.4 Evaluation of Sparse Latent Representations

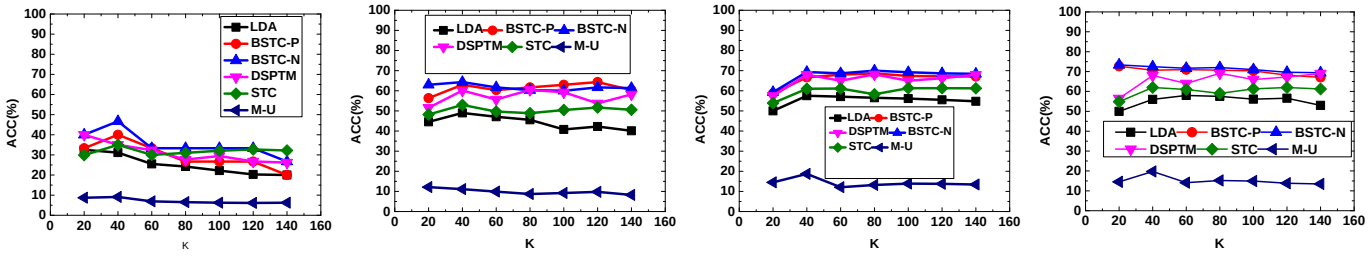
In this part, we further compare the sparsity of the learned latent representations of words, topics and documents from different models qualitatively.

Word codes: We further quantitatively evaluate the sparsity of learned word codes. Figure 10a presents the average word representations of top-representation words learned by *BSTC-P* of three example categories in training documents. We calculate average word codes for all documents in each category. Here, we only display the results of *BSTC-P* to save space. *BSTC-P* tends to learn a narrow spectrum of topics for each word. It is obvious that there are only little non-zero elements in learned average word codes, and the little non-zero elements have significant weights. These illustrate *BSTC-P* can learn quite clear and sparse word-level representations. We further quantitatively evaluate the sparsity of the learned word codes. Figure 11 presents the average word sparse ratio of 20NG. We can see the average word codes learned by *BSTC-P* and *BSTC-N* are sparser than those learned by other three models, and the four sparsity-enhanced models outperform *LDA*, which have no mechanism to induce word-level sparsity. As for *BSTC-P* and *BSTC-N*, the *BSTC-N* outperforms *BSTC-P*, because the Normal-Jeffrey prior of *BSTC-N* has more chance to lead a sparse solution than the Laplace hierarchical prior of *BSTC-P*. All these above proves the hierarchical sparse prior can induce sparser representations than traditional lasso of *STC*, as well as sparse group lasso of *STCSG*, and it can also induce sparser representations than hierarchical Laplace sparse prior of *BSTC-P*.

Topic codes: Similar to word codes, we also display the average topic code of the most possible topic learned by *BSTC-P* for three example categories in training documents in Figure 12. As we can see, the representations of three example topics only focus on a small set of words and all of non-zero elements are salient and distinct. Moreover, we quantitatively evaluate the average sparse ratio of topic codes in Figure 10b. Not surprisingly, we can find that the sparse ratios of *BSTC-P*, *BSTC-N* and *STCSG* are much higher than those of *STC* and *LDA*, which nearly have no direct sparsity control over topic representations. Meanwhile, the sparse ratio of *STC* is also higher than that of *LDA*. This is mainly because that *STC* has direct control over word and document topic representations, but not for *LDA*. The topic codes learned by *BSTC-P* and *BSTC-N* are sparser than those learned by *STCSG*, which indicates that imposing the hierarchical sparse prior on word codes makes better performance than imposing non-hierarchical sparse prior. As for *BSTC-P* and *BSTC-N*, *BSTC-N* outperforms *BSTC-P* again, mainly because imposing the Normal-Jeffery sparse prior on word codes.

Document codes: Figure 13 shows the average document codes for 3 example categories discovered by *BSTC-P*, unlike *STC*, we find *BSTC-P* tends to learn a narrow spectrum of topics, and can obtain discriminative and sparse representations of documents. It is also worth noting that,

5. http://www.csie.ntu.edu.tw/ cjlin/libsvm/



(a) 20NG_0.2%. (b) 20NG_1%. (c) 20NG_10%. (d) 20NG_100%.

Fig. 8: Classification accuracy on 20NG.

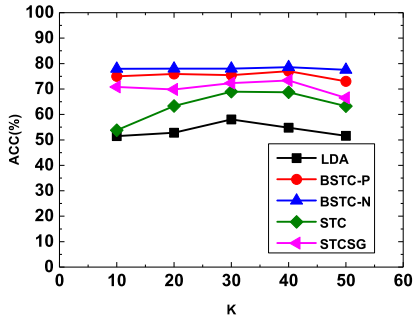


Fig. 9: Classification accuracy on Twitter.

although each document code can be sparse, the average document codes seems to be not so sparse. Moreover, Figure 10c presents the sparse ratio of document codes with setting different topic numbers. We observe that *BSTC-P* and *BSTC-N* outperform *STCSG* and *STC*, which indicates that directly imposing hierarchical sparse prior can achieve better sparsity again. Meanwhile, the sparse ratio of *BSTC-P* is lower than that of *BSTC-N*, which also proves that the Normal-Jeffrey hierarchical sparse prior can make better performance in sparsity than the Gamma-Jeffrey sparse prior.

6 Conclusion

In this paper, we have presented a novel topic model, called Bayesian sparse topical coding with Poisson distribution (*BSTC-P*) based on our recent work of Sparse Topical Coding with Sparse Groups (*STCSG*). In this model, the sparse Bayesian learning was introduced to improve the learning of sparse word, topic and document representations. *BSTC-P* gains advantages in learning word, topic and document proportions. Meanwhile, a sparsity-enhanced version of BSTC is also proposed to obtain the sparsest optimal solution by putting the Normal-Jeffrey hierarchical prior. The Expectation Maximization (EM) and the Variational Inference procedure are incorporated to learn *BSTC-P* and *BSTC-N* effectively. Experimental results shows that both methods can achieve better performance than other baseline approaches in learning meaningful and sparse representations of texts, thus accordingly improve the document classification accuracy. As a part of our future work, we will explore the suitable number of initial keyword sets number, and investigate more efficient algorithms for *BSTC-P*.

Acknowledgments

The work is supported by the National Science Foundation of China (NSFC, No. 61472291 and No. 41472288)

References

[1] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent dirichlet allocation, the Journal of machine Learning research 3 (2003) 993–1022.

[2] D. M. Blei, J. D. Lafferty, Dynamic topic models, in: Proceedings of the 23rd international conference on Machine learning, ACM, 2006, pp. 113–120.

[3] L. AlSumait, D. Barbará, C. Domeniconi, On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking, in: Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on, IEEE, 2008, pp. 3–12.

[4] X. Wang, A. McCallum, Topics over time: a non-markov continuous-time model of topical trends, in: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2006, pp. 424–433.

[5] N. Simon, J. Friedman, T. Hastie, R. Tibshirani, A sparse-group lasso, Journal of Computational and Graphical Statistics 22 (2) (2013) 231–245.

[6] T. Eltoft, T. Kim, T.-W. Lee, On the multivariate laplace distribution, Signal Processing Letters, IEEE 13 (5) (2006) 300–303.

[7] M. Heiler, C. Schnörr, Learning sparse representations by non-negative matrix factorization and sequential cone programming, The Journal of Machine Learning Research 7 (2006) 1385–1407.

[8] L. Bai, J. Guo, Y. Lan, X. Cheng, Group sparse topical coding: from code to topic, in: Proceedings of the sixth ACM international conference on Web search and data mining, ACM, 2013, pp. 315–324.

[9] J. Zhu, E. P. Xing, Sparse topical coding, arXiv preprint arXiv:1202.3778.

[10] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society. Series B (Methodological) (1996) 267–288.

[11] C. Wang, D. M. Blei, Decoupling sparsity and smoothness in the discrete hierarchical dirichlet process, in: Advances in neural information processing systems, 2009, pp. 1982–1989.

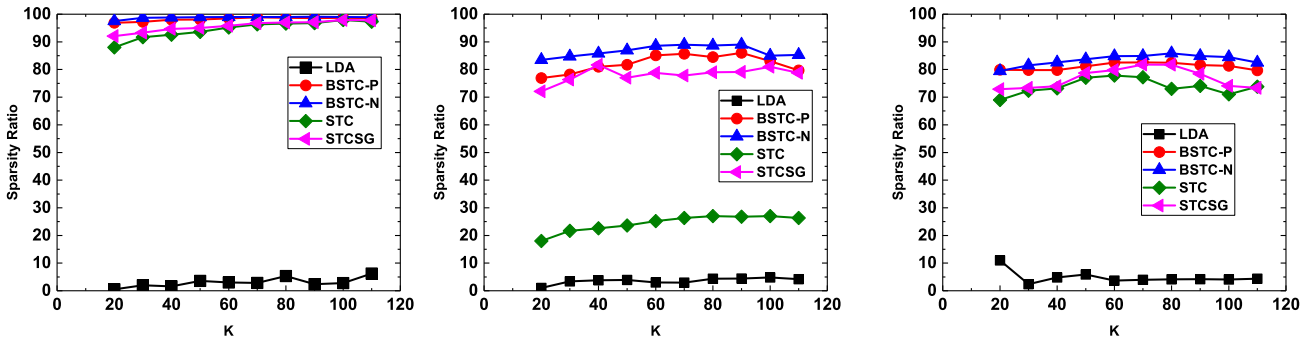
[12] T. Lin, W. Tian, Q. Mei, H. Cheng, The dual-sparse topic model: mining focused topics and focused terms in short text, in: Proceedings of the 23rd international conference on World wide web, ACM, 2014, pp. 539–550.

[13] J. Huang, M. Peng, H. Wang, J. Cao, W. Gao, X. Zhang, A probabilistic method for emerging topic tracking in microblog stream, World Wide Web (2016) 1–26.

[14] K. Than, T. B. Ho, Fully sparse topic models, in: Machine Learning and Knowledge Discovery in Databases, Springer, 2012, pp. 490–505.

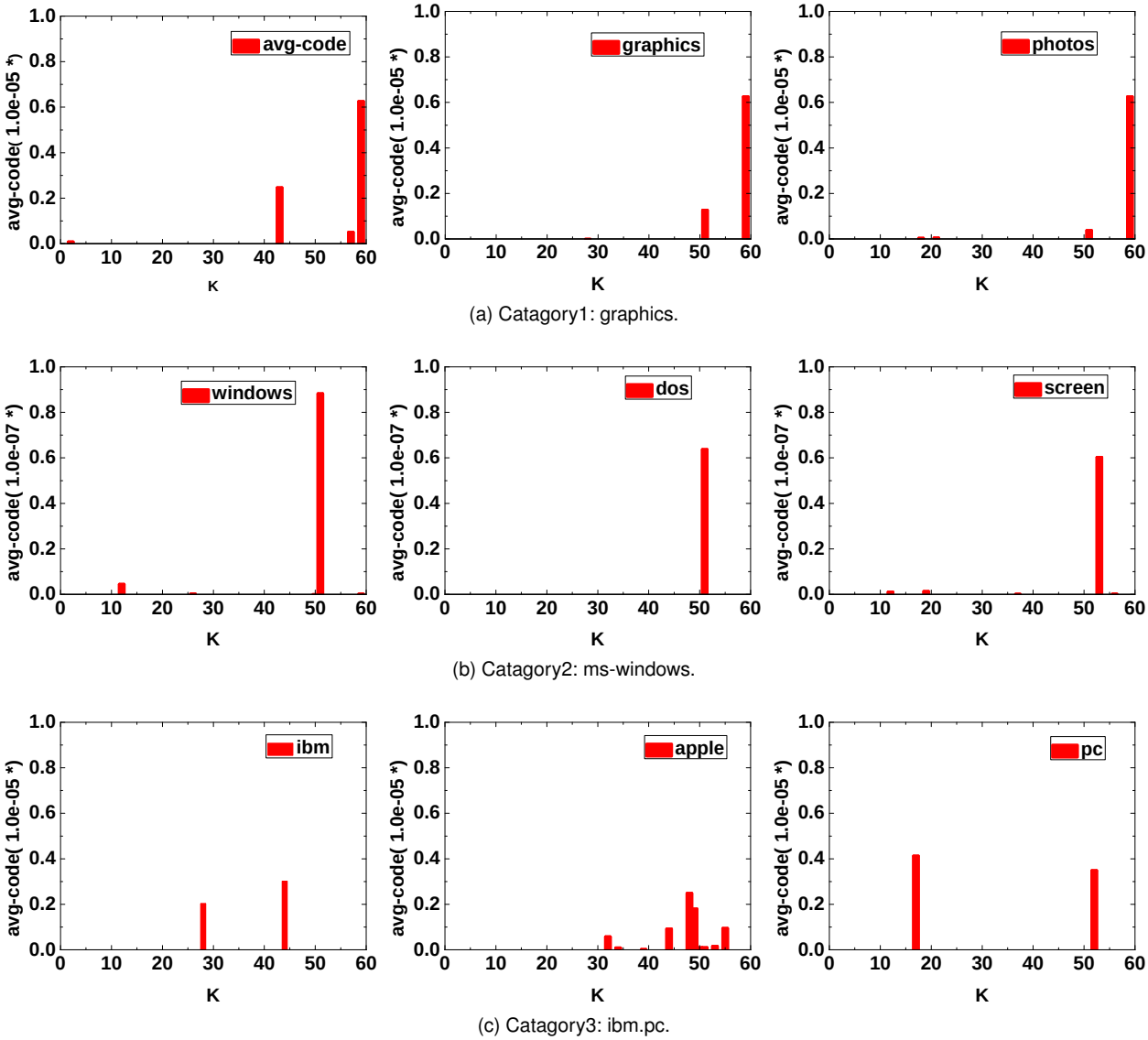
[15] T. Hofmann, Probabilistic latent semantic indexing, in: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 1999, pp. 50–57.

[16] M. Peng, Q. Xie, J. Huang, J. Zhu, S. Ouyang, J. Huang, G. Tian, Sparse topical coding with sparse groups, in: International Conference on Web-Age Information Management, Springer, 2016, pp. 415–426.



(a) Average sparse ratio of word codes (b) Average sparse ratio of topic codes (c) Average sparse ratio of document codes

Fig. 10: The sparse ratio of a latent semantic representing



(a) Catagory1: graphics.

(b) Catagory2: ms-windows.

(c) Catagory3: ibm.pc.

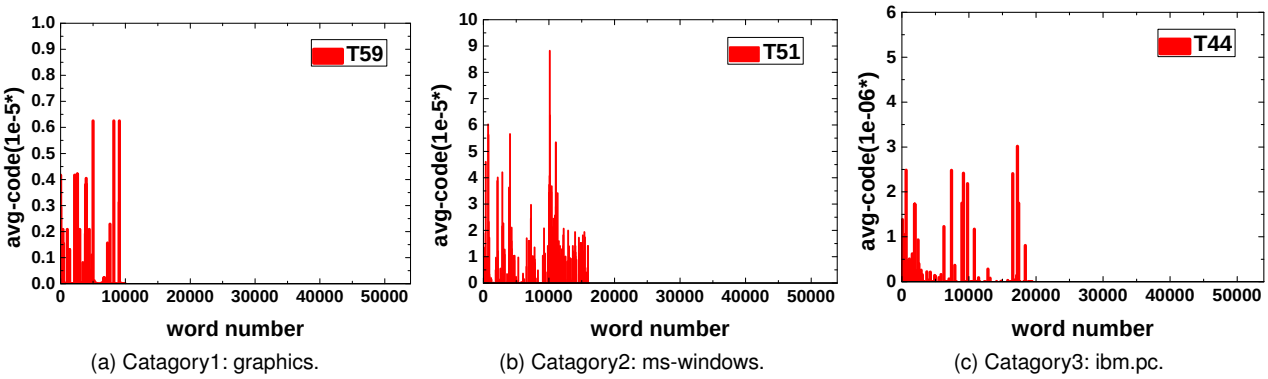
Fig. 11: The average word codes of top-representation words learned by *BSTC-P* of three example categories.

[17] J.-T. Chien, Y.-L. Chang, Bayesian sparse topic model, Journal of Signal Processing Systems 74 (3) (2014) 375–389.

[18] A. T. Cemgil, Bayesian inference for nonnegative matrix factorisation models, Computational Intelligence and Neuroscience 2009.

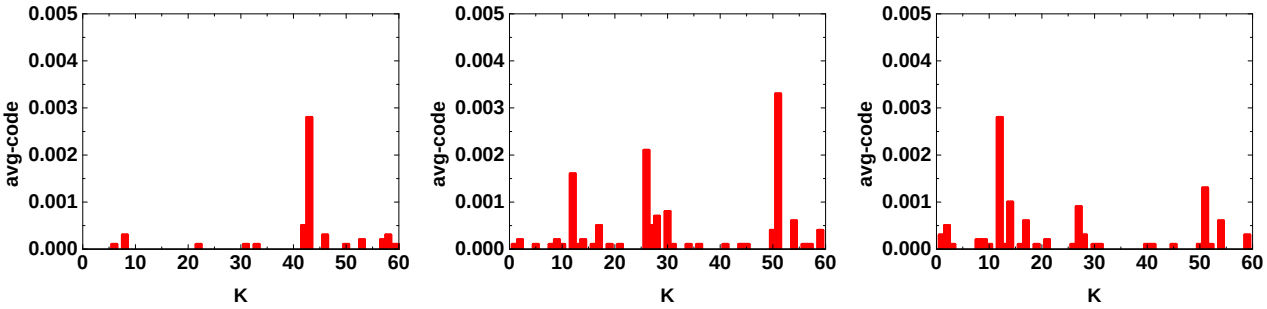
[19] J. Paisley, D. Blei, M. I. Jordan, Bayesian nonnegative matrix factorization with stochastic variational inference, Handbook of Mixed Membership Models and Their Applications. Chapman and Hall/CRC.

[20] T. Sakaki, M. Okazaki, Y. Matsuo, Earthquake shakes twitter users: real-time event detection by social sensors, in: Proceedings of the



(a) Catagory1: graphics. (b) Catagory2: ms-windows. (c) Catagory3: ibm.pc.

Fig. 12: The average topic codes of the most possible topic learned by *BSTC-P* of three example categories.



(a) Catagory1: graphics. (b) Catagory2: ms-windows. (c) Catagory3: ibm.pc.

Fig. 13: The average document codes learned by *BSTC-P* of three example categories.

19th international conference on World wide web, ACM, 2010, pp. 851–860.

[21] K. Sasaki, T. Yoshikawa, T. Furuhashi, Online topic model for twitter considering dynamics of user interests and topic trends., in: EMNLP, 2014, pp. 1977–1985.

[22] M. Peng, J.-J. Huang, N. Ghani, S.-T. Sun, B. Wu, Y.-X. He, W.-D. Wen, Micro-blogger influence analysis based on user features, ŁŁ 14 (2) (2013) 307–314.

[23] F. Doshi-Velez, B. Wallace, R. Adams, Graph-sparse lda: a topic model with structured sparsity, arXiv preprint arXiv:1410.4510.

[24] M. Yuan, Y. Lin, Model selection and estimation in regression with grouped variables, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68 (1) (2006) 49–67.

[25] S. Williamson, C. Wang, K. Heller, D. Blei, Focused topic models, in: NIPS Workshop on Applications for Topic Models: Text and Beyond, 2009.

[26] S. Williamson, C. Wang, K. A. Heller, D. M. Blei, The ibp compound dirichlet process and its application to focused topic modeling, in: Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 1151–1158.

[27] X. Chen, M. Zhou, L. Carin, The contextual focused topic model, in: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2012, pp. 96–104.

[28] C. Archambeau, B. Lakshminarayanan, G. Bouchard, Latent ibp compound dirichlet allocation, IEEE transactions on pattern analysis and machine intelligence 37 (2) (2015) 321–333.

[29] J. Kujala, Sparse topic modeling with concave-convex procedure: Flemish algorithm for latent dirichlet allocation, Tech. rep., Technical Report, 2004. 2 (2004).

[30] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (6755) (1999) 788–791.

[31] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, R. Harshman, Indexing by latent semantic analysis, Journal of the American society for information science 41 (6) (1990) 391.

[32] M. E. Tipping, Sparse bayesian learning and the relevance vector machine, Journal of machine learning research 1 (Jun) (2001) 211–244.

[33] D. P. Wipf, B. D. Rao, Sparse bayesian learning for basis selection, IEEE Transactions on Signal Processing 52 (8) (2004) 2153–2164.

[34] S. D. Babacan, R. Molina, A. K. Katsaggelos, Bayesian compressive sensing using laplace priors, IEEE Transactions on Image Processing 19 (1) (2010) 53–63.

[35] Z. Zhou, K. Liu, J. Fang, Bayesian compressive sensing using normal product priors, IEEE Signal Processing Letters 22 (5) (2015) 583–587.

[36] M. A. Figueiredo, Adaptive sparseness for supervised learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (9) (2003) 1150–1159.

[37] M. Kyung, J. Gill, M. Ghosh, G. Casella, et al., Penalized regression, standard errors, and bayesian lassos, Bayesian Analysis 5 (2) (2010) 369–411.

[38] D. L. Donoho, M. Elad, Optimally sparse representation in general (nonorthogonal) dictionaries via 1 minimization, Proceedings of the National Academy of Sciences 100 (5) (2003) 2197–2202.

[39] S. N. David Wipf, Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions, Journal of Selected Topics in Signal Processing (Special Issue on Compressive Sensing) 4 (2) (2010) 317–329.

[40] E. J. Candes, M. B. Wakin, S. P. Boyd, Enhancing sparsity by reweighted 1 minimization, Journal of Fourier analysis and applications 14 (5-6) (2008) 877–905.

[41] M. N. Schmidt, H. Laurberg, Nonnegative matrix factorization with gaussian process priors, Computational intelligence and neuroscience 2008 (2008) 3.

[42] H. Lee, R. Raina, A. Teichman, A. Y. Ng, Exponential family sparse coding with application to self-taught learning., in: IJCAI, Vol. 9, Citeseer, 2009, pp. 1113–1119.

[43] W. Buntine, A. Jakulin, Discrete Component Analysis, Springer Berlin Heidelberg, 2006, pp. 1–33.

[44] H. Zou, T. Hastie, R. Tibshirani, Sparse principal component analysis, Journal of computational and graphical statistics 15 (2) (2006) 265–286.

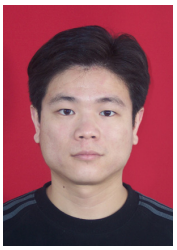
[45] M. J. Wainwright, M. I. Jordan, Graphical models, exponential

families, and variational inference, Foundations and Trends® in Machine Learning 1 (1-2) (2008) 1–305.
[46] M. D. Hoffman, D. M. Blei, C. Wang, J. W. Paisley, Stochastic variational inference., Journal of Machine Learning Research 14 (1) (2013) 1303–1347.
[47] J. Hou, Y. Zhang, Effectively finding relevant web pages from linkage information, IEEE Transactions on Knowledge and Data Engineering 15 (4) (2003) 940–951.
[48] T. K. Moon, The expectation-maximization algorithm, IEEE Signal processing magazine 13 (6) (1996) 47–60.
[49] X. Pu, R. Jin, G. Wu, D. Han, G.-R. Xue, Topic modeling in semantic space with keywords, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, ACM, 2015, pp. 1141–1150.
[50] Y. Gu, G. Liu, J. Qi, H. Xu, G. Yu, R. Zhang, The moving k diversified nearest neighbor query, IEEE Transactions on Knowledge and Data Engineering 28 (10) (2016) 2778–2792.
[51] Y. Guan, J. G. Dy, Sparse probabilistic principal component analysis., in: AISTATS, 2009, pp. 185–192.
[52] S. Chen, D. Donoho, Basis pursuit, in: Signals, Systems and Computers, 1994. 1994 Conference Record of the Twenty-Eighth Asilomar Conference on, Vol. 1, IEEE, 1994, pp. 41–44.
[53] J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties, Journal of the American statistical Association 96 (456) (2001) 1348–1360.
[54] A. E. Hoerl, Application of ridge analysis to regression problems, Chemical Engineering Progress 58 (3) (1962) 54–59.
[55] D. Newman, J. H. Lau, K. Grieser, T. Baldwin, Automatic evaluation of topic coherence, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2010, pp. 100–108.
[56] P. Pecina, Lexical association measures and collocation extraction, Language resources and evaluation 44 (1-2) (2010) 137–158.
[57] J. Tang, Z. Meng, X. Nguyen, Q. Mei, M. Zhang, Understanding the limiting factors of topic modeling via posterior contraction analysis, in: ICML, 2014, pp. 190–198.
[58] M. Peng, B. Gao, J. Zhu, J. Huang, M. Yuan, F. Li, High quality information extraction and query-oriented summarization for automatic query-reply in social network, Expert Systems with Applications 44 (2016) 92–101.

Min Peng is a professor in Computer School at Wuhan University, Wuhan, China. She received her Ph.D. degree in Computer Software and Theory from Wuhan University in 2006. She worked as a post-doctor from 2009 to 2010 at the Advanced Cyber-Infrastructure Laboratory, ECE Department at the University of New Mexico, USA. Her current research focus areas include information retrieval, network services and natural language process.

Qianqian Xie is a Ph.D candidate in Computer School at Wuhan University, Wuhan, China. She received her bachelors degrees from Jiangxi Normal University. Her current research focus areas include topic model, sparse coding and deep learning.

Yanchun Zhang is a professor and director of the Centre for Applied Informatics (CAI) in Victoria University. His current research interests include databases, data mining, health informatics, web information systems, and web services (yanchun.zhang@vu.edu.au).



Gang Tian is a lecturer and PhD in Wuhan University. His research focus is in the fields of Big Data Techniques and Mining, machine vision, and machine learning (tiang2008@whu.edu.cn).